DOCUMENT RESUME

ED 249 598 EA 017 123

AUTHOR Linn, M. C.; Fisher, C. W.
TITLE The Gap between Promise and Reality in Computer Education: Planning a Response.
PUB DATE Jun 84
NOTE 21p.; In: "Making Our Schools More Effective: Proceedings of Three State Conferences." See EA 017 101.
PUB TYPE Speeches/Conference Papers (150) -- Viewpoints (120)

EDRS PRICE MF01/PC01 Plus Postage.
DESCRIPTORS Cognitive Development; *Computer Assisted Instruction; Computer Oriented Programs; *Computers; Computer Software; *Curriculum Development; Elementary Secondary Education; Inservice Teacher Education; *Microcomputers; *Planning; *Program Development
IDENTIFIERS Computer Anxiety; Computer Uses in Education

ABSTRACT

This paper identifies several reasons for the gap between the possible instructional uses of computers and their actual use in classrooms and discusses considerations in planning programs which will bring computer education closer to its envisioned promise. The Assessing the Cognitive Consequences of Computer Environments for Learning (ACCCEL) project is briefly described and six key features which distinguish the computer as a facilitator of cognitive skills (interactiveness, precision, consistency, challenge, complexity, and provision for multiple solutions) are presented based on the project's preliminary investigations. Impediments to effective planning for computer education are also discussed. These include (1) lack of control of hardware decisions; (2) lack of available, effective educational software; (3) lack of adequate computer-based curricula; and (4) lack of appropriate training for program implementers. A discussion is presented of school-based planning for computer decision making, including innovative programs at various schools, investigations conducted by research groups, and insights shared by thoughtful teachers. Four major areas of concern are covered: planning for cognitive outcomes, planning for affective consequences, planning for equitable outcomes, and planning for teacher professional development. (TE)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

THE GAP BETWEEN PROMISE AND REALITY IN COMPUTER EDUCATION: PLANNING A RESPONSE

M. C. Linn
University of California, Berkeley

C. W. Fisher
Far West Laboratory

U.S. DEPARTMENT OF EDUCATION
NATIONAL INSTITUTE OF EDUCATION
EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)
X This document has been reproduced as received from the person or organization originating it.
Minor changes have been made to improve reproduction quality.
• Points of view or opinions stated in this document do not necessarily represent official NIE position or policy.

ED249598

The current gap between the promise of computers in schools and the reality of their use in classrooms is simply too great. Before a national magazine heralds the trend with a cover story such as "Macro Expectations, Micro Outcomes: Computers Fail School," a concerted effort is needed to change the situation. Computer education can and should live up to many of the promises which have been made.

In this paper we identify several reasons for this "gap" and discuss factors schools might consider in planning programs which will bring computer education closer to the envisioned promise of computers in schools. We focus on the instructional uses of computers rather than on the managerial and other supporting uses also prevalent in schools today. The gap between the possible instructional uses of computers in education and their actual use results from a variety of reasons, many of them completely beyond the control of schools. In this paper, we focus on factors within the control of schools and reflected in school-based decision making. We draw on experiences that schools have had using innovative programs, on investigations conducted by research groups, and on insights shared by thoughtful teachers. We suggest how schools can use what is known about computers in education to design programs to meet their needs.

Project ACCCEL

The Assessing the Cognitive Consequences of Computer Environments for Learning (ACCCEL) project, in a series of investigations conducted at the Lawrence Hall of Science and the Far West Laboratory, seeks ways to increase the effectiveness of computers in schools. This project, funded by the National Institute of Education, is evaluating the cognitive consequences of computer-based experiences on students and has focused primarily on examining experiences which have the greatest potential for influencing student cognitive performance. (Additional information about the ACCCEL program is available from the authors.) The preliminary investigations of the ACCCEL project offer the beginnings of a model of computer-based instruction.

EA 017 123

The Promise of Computer Learning Environments

Promises have been made for computer learning environments because of their great potential for fostering higher cognitive skills. As the ACCCEL project has clarified, and is discussed further below, computers can foster skills such as planning problem solutions and

debugging the proposed plan. Schools frequently point out that they wish to teach these skills.

A variety of national commissions, such as the National Commission on Excellence in Education and the National Science Board, have recently drawn attention to the deplorable state of education in our nation's schools and have called for educational innovations which encourage these skills variously referred to as the "new basics," problem solving and a "learning society." If used effectively, computer learning environments have potential for responding to this call.

The computer is a tool like the printing press or the pencil. The potential uses of the tool are great. However, serious effort and substantial resources are needed to design programs and train people who can capitalize on the potential of the tool. Six key features of the computer learning environment distinguish it from other learning environments and make it an exceptional facilitator for the acquisition of cognitive skills: (a) interactiveness, (b) precision, (c) consistency, (d) challenge, (e) complexity, and (f) provision for multiple solutions. Each of these features is examined in turn.

First, computer learning environments are interactive. Whether one is using a computer simulation, playing a computer game, or developing a computer program, there is a cyclical process of providing information to and receiving feedback from the computer. Although the time required for the response-feedback cycle can vary greatly, all computer learning environments are characterized by interactiveness. To proceed with any computer task, the learner must respond actively. That is, the learner must submit another program, initiate another keystroke, or move the paddle control. The requirement that the learner respond periodically (usually very quickly) insures some level of cognitive activity.

The quality of the cognitive activity generated in computer learning environments is dependent upon the software or programming task that is used. Although interactiveness is not unique to computer learning environments, the degree to which computer-learner interactions can be controlled and potentially used for increasing learning efficiency is an important characteristic of computer learning environments. In addition, computer learning environments are far more likely to provide rapid feedback than are other classroom environments where assignments may be returned days or weeks after.

Second, the computer learning environment is precise. To communicate with the machine, the learner must specify his message completely. In general, the computer does not make inferences, "read between the lines," or otherwise use implicit knowledge to interpret the desires of the learner. This explicitness requires the learner to differentiate among items that previously were undifferentiated, or to "think through" aspects of a problem that previously were unclear, and to give relatively complete specification of messages.

Thus, the preciseness of computer learning environments encourages students to be precise.

The computer response can also be precise if properly programmed. The precise information provided in program debugging often exceeds that in other learning environments. Rather than just saying an effort to solve a problem is wrong, the computer can diagnose some aspects of what might be wrong.

Third, computer learning environments are consistent. This consistency is manifested in several ways. If the learner provides information to the computer (in the form of a program, or a series of keystrokes), the computer will respond in a specific and concrete manner. If the learner repeats his action (resubmits the program or repeats the same series of keystrokes), the computer will give the exact same response. If the learner does the same thing, the computer will do the same thing.

The computer learning environment is consistent in a second important way. The computer will provide a consistent response (to the same input) regardless of who the learner/programmer is. An obese or a thin person, a tall or short person, a native English-speaker or a native Spanish-speaker, a black-skinned or brown-skinned person, all will get the same response from the computer, provided the same information was "fed in." That is, the feedback provided to learners does not vary with "irrelevant" characteristics of the learner. These, more or less obvious, points about the consistency of the computer learning environments have important implications for what and how much students learn since the feedback should be easier to identify, compared to other environments where the feedback is less than consistent. Consistent feedback can make learning efficient.

Fourth, the computer provides a challenging and motivating environment in which to learn. Many teachers have observed the motivational qualities of the computer in the classroom. Proponents of the use of games and simulations in learning environments state that games are good problem-solving environments that stimulate exploration, discovery learning, and formulation of strategies.

Fifth, computer learning environments are complex. Computers, computer systems, and software are multi-functional in their operation. This complexity can encourage or require higher cognitive processing. For example, the computer can require the learner to build a conceptual representation (mental model) of its operation in order to solve problems. The process of building a mental model requires several important cognitive skills. First, one must be able to organize many detailed, low-level facts into a coherent knowledge structure. Second, one must predict the operation of the computer on the basis of the model. This entails accessing one's knowledge representation in the act of writing a program. Third, one must compare the predicted with actual outcomes. Sometimes, such comparisons require interpretation of empirical evidence.

Fourth, one must diagnose the cause of discrepancies between expected and actual outcomes. Lastly, one must refine one's knowledge so that the mental model is consistent with the actual system. This process is interactive in nature, in that the cycle of prediction, testing, and refinement is repeated until an accurate model is constructed. Compared to other approaches, the computer provides a safe, cheap, and fast environment for learning a complex system (e.g., compare to costs of experimenting with nuclear reactors).

Sixth, computer learning environments provide for the presentation of problems to learners that have many correct solutions. Multiple solutions encourage divergent thinking, creative approaches to problem solving, individual expression and style, consideration of varied alternatives, and subjective judgment of appropriateness and elegance.

Schools can capitalize on these six features of the computer learning environment to foster higher cognitive skills, to achieve affective outcomes, to increase equity of educational opportunity, and to design effective teacher professional development programs. The computer provides a unique tool for schools; planning is required to maximize its power. Using computers as smart typewriters or as automated flash cards, while helpful, fails to take advantage of their full potential.

Taking Advantage of the Computer Environment: School-Based Decision Making About Computer Education

In the area of computer education, schools frequently find that realistic planning is, at best, frustrating. Even after expending much effort at the school and the district level, programs frequently appear haphazard or unsuccessful. Impediments to effective planning for computer education include (a) lack of control of hardware decisions, (b) lack of available, effective software to accomplish the desired objectives of computer education programs, (c) lack of curricula which incorporate computer-based activities, and (d) lack of appropriate training for program implementers.

Although few school districts have been able or willing to design long-range strategic plans which incorporate what is known about computers into a program which can maximize the outcomes from computer education, the opportunity exists. Before we find that computers have been placed in closets and are collecting dust, we need to capitalize on what we do know about computers in education. By maximizing the effectiveness of existing programs, we can narrow the gap between promise and reality in computer education.

Hardware Selection

Many factors contribute to haphazard planning in schools. A large number of these factors are absolutely beyond the control of

those making the decisions in schools. For example, computer hardware is frequently donated by industry, by parent groups, and by business consortia. Sometimes schools are simply not consulted about these decisions before the hardware arrives. Funding of computer education programs is often unpredictable. State programs for computer education may suddenly emerge and equally suddenly be discontinued. The federal government has increased and decreased funding for a variety of programs that directly and indirectly influence computer-based learning experiences such as equity programs, vocational training programs, and programs for the learning disabled.

The "buy now, plan later" approach to hardware selection frequently appears to be unavoidable. Nevertheless, districts which buy computer hardware now and come up with an instructional plan later are likely to widen the gap between the promise of computers in education and the reality of their use in the classroom. Planning is necessary for equitable and exciting outcomes from computer-based instructional programs. Haphazard approaches to computer-based instruction result in haphazard outcomes and leave schools open to severe criticism about the effectiveness of their computer-based instructional program.

A misplaced perception that "if you don't have a product for your expenditures, you're wasting your money," may permeate some of the decision making about instructional programs using computers. Districts may purchase hardware before adequate planning for appropriate teacher training because hardware results in a physical product which can be shown to the school board, visiting parents, and other concerned individuals. In the long run, the presence of hardware in schools could become a point of embarrassment rather than a point of pride. Before such a situation emerges, planning for effective outcomes from computer-based learning environments is required.

Schools often have the opportunity to influence the hardware selection process. Now that numerous districts have computers, it is possible to learn from the experiences of others. Clearly, schools will make more effective decisions if they have some idea about what they want the hardware to do for them, hence the need for planning. On the other hand, as discussed below, the potential uses of computers have not been fully explored so schools can expect to modify their plans frequently as new information emerges and as they learn more about the skills of their teachers and the capabilities of their students. Planning for effective hardware selection requires on-going information gathering and flexible adaptation to new situations.

Software Selection

Software selection is intertwined with hardware selection since educational software is often hardware-dependent. On-going decision making about software selection is especially important for effective computer education. New products appear on the market constantly. A major impediment to effective software selection is lack of infor-

mation about new developments. Several factors contribute to this situation.

First, the dissemination of information about educational software is poorly established. It is difficult for schools to find out about new products and frequently the only source of information is a salesperson for a particular product rather than a network or consortium of individuals who have reviewed the range of products. The sales people who generally provide information about innovation in the field of computer education have vested interests in selling their own products. A few groups such as Computer Using Educators (CUE) offer regional conferences where information can be shared. It should be noted, however, that school district budgets do not always provide funds for teachers to attend such conferences. Some states have technical centers which gather information, and several clearinghouses for educational software are now emerging, such as EPIE, funded by the National Institute of Education and located in New York, and the Minnesota Educational Computing Consortium (MECC), but more comprehensive efforts are needed.

Second, teachers have not yet developed the ability to browse through educational software in order to select software which will meet their needs. Over the years we have developed techniques for browsing through books. However, browsing through software requires a different set of skills and is frequently quite ineffective. For example, Rocky's Boots, a program which many teachers feel is one of the best examples of a learning environment which demands strategic planning and debugging, is frequently rated poorly by teachers who are given only a short period of time to examine the software. This situation results because teachers have difficulty in distinguishing the cognitive learning features of the software from the logistic features. One important aspect of teacher/professional development must be training on criteria which are appropriate for making software selection.

Third, relatively little software has been developed for the educational market when compared to the entertainment market. Furthermore, educational software is frequently targeted to the home rather than to the school, largely because the education market is thin and unpredictable and because dissemination to schools is difficult.

Much more cognitively interesting and demanding software is needed in order for schools to be able to capitalize on the computer learning environment. Given the dearth of such software, schools are teaching programming more frequently than they probably would if other software were available which stressed problem-solving skills. Programming has the advantage of providing the problem-solving opportunities and the disadvantage of not emphasizing subject matter for a specific course. Ultimately, cognitively interesting software for computer-based learning environments will emphasize problem solving using the subject matter in the particular course for which the software is used.

Fourth, distribution of software to schools is difficult. At the present time, textbook companies have the major network for distributing information to schools. When textbook companies do select software for distribution, they frequently make the same sorts of errors which are made by others unfamiliar with browsing through software. They select software which looks more like books and is less likely to take advantage of the potential of a computer learning situation.

Fifth, schools often select software because it is inexpensive or free at a software exchange rather than because it is the best. Such decision making discourages software developers. The best software takes years to develop and requires much evaluation and revision. Quickly developed tools frequently feature either drill and practice or display of information. These programs often fail to use the computer environment to its full potential and ultimately convince observers that the computer is not effective for schools.

These factors work together to discourage the selection of effective educational software. Since distribution networks are poorly developed, manufacturers are not motivated to develop products for the educational market. Since selection of software requires training and experience and since information about software is difficult to acquire, schools are tempted to make haphazard decisions.

Computer Curricula

A major impediment to effective use of computers in schools is the lack of curricula which incorporate computer-based activities. When software is available it is rarely incorporated into a curriculum. An exception is the Creative Play curriculum for upper elementary school available from the Lawrence Hall of Science, Berkeley, CA.

When curricula are available they often emphasize drill and practice rather than the higher cognitive skills which computers are capable of exercising (as discussed in the next section). Very few textbooks emphasize the higher order problem-solving skills which make the computer learning environment especially effective. Indeed, explicit instructions for higher cognitive skills such as planning and debugging programs are frequently lacking in curriculum materials, even curriculum materials for programming taught in colleges. A recent survey of textbooks for college use reveals that less than one percent of textbook pages are devoted to debugging, an activity which frequently comprises 50 percent of a programmer's time. Thus, curriculum materials have not adequately responded to the nature of the computer learning environment. The lack of effective curriculum materials greatly inhibits the possibility of making effective use of this environment.

Lack of effective curricula for computer education places major responsibility on school-based decision makers for selecting software to complement the educational program. Even if schools have decided

what they want the computer education program to accomplish, they have difficulty ensuring that the materials they select will do the job.

Staff Training

The lack of appropriate training for program implementation is another impediment to successful operation of computer-based instructional programs. The time required for teachers to develop their own computer skills is often underestimated. Since most schools have begun to implement instruction in computer programming, there has been considerable demand for programming instructors. In many cases, staff for programming courses come from the ranks of experienced classroom teachers who have little previous experience in programming. It is not unusual for programming instructors in some schools to have only six months of programming experience when they begin to teach programming. The time and support required to provide appropriate training for instructors present educational decision makers with some difficult problems. Since schools are under considerable pressure to provide programming instruction, they must respond relatively quickly, often before they have had time to develop and implement high quality training procedures for the programming instructors. Two factors prevent schools from attracting competent programmers to teaching from other sectors of the economy. First, competent programmers have higher income levels than teachers. In fact, the schools may be in some danger of losing their best programming teachers to more high-paying jobs outside of school. Second, the schools are, in many cases, looking for new instructional areas like programming to absorb experienced teachers who have been displaced by lower enrollments and school closings.

Although the training of programming instructors poses a problem for schools, this problem is not insurmountable since it involves a relatively small number of teachers. A larger and more serious problem arises when we consider the integration of computer-based activities into the mainstream of the school curriculum. If schools are to take advantage of the computer learning environment by incorporating computer-based simulations and other activities (other than programming per se) in reading, writing, mathematics, science, social studies, and art instruction, then some computer-related training will be required by the vast majority of practicing teachers and virtually all prospective teachers. The sheer numbers of people to be trained and the great diversity in the applications that teachers will have to be familiar with, make the staff development effort very complex. The general issue of staff training has wide ranging implications for the scope and sequence of any computer-based instruction and should be one of the major considerations in program planning.

School-Based Planning for Computer Decision Making

There is no question about the difficulty of school-based planning for computer education. The gap between promise and reality in computer education stems from many factors which make planning difficult. Nevertheless, experiences of schools using computers, research on computer learning environments, and reflections of expert teachers offer insight into four major questions facing school-based planners. These questions are listed in Table 1 and discussed subsequently.

Table 1

School-Based Computer Education Decision Making:
Questions to Consider

1. What cognitive outcomes from computer education programs are possible and desired?
2. What affective outcomes are possible and desired?
3. How can equitable access to computers be ensured?
4. How can professional staff development be fostered?

Planning for Cognitive Outcomes

Computer learning environments have the potential for fostering higher cognitive skills including strategic planning skills and debugging skills. By strategic planning we mean the ability to combine diverse pieces of information to generate a unique problem solution; by debugging we mean the ability to diagnose and remediate difficulties in a strategic plan. These skills are objectives of most educational programs, yet, the difficulty of teaching them often results in only token consideration in curriculum efforts. Computer environments for learning can greatly intensify the opportunity to focus on strategic planning and debugging in educational programs.

Although strategic planning and debugging are the ultimate objectives of a large number of computer-based learning programs, frequently the programs are neither long enough nor intensive enough to achieve these consequences. On the other hand, many programs involve drill and practice rather than these more complex activities. Programs which emphasize drill and practice often fail to focus on any higher cognitive skills.

It is possible to define a chain of cognitive consequences from the direct outcomes of initial computer activities to more general strategic planning and debugging skills. Educational programs which move students along this chain will culminate in important cognitive outcomes.

Instruction in computer learning environments will be especially effective if the chain of cognitive consequences is clearly delineated so that preliminary experiences are designed with the ultimate outcomes

in mind. Instruction which stresses the cognitive outcomes most likely to result from computer learning environments will intensify the experience and increase the likelihood that students will gain cognitively useful information from the learning experience.

Currently, the most cognitively demanding classroom use of the computer is probably for learning to program. Activities which lead to programming expertise can form a chain of cognitive consequences from computer learning. Skilled programmers use strategic planning and debugging skills extensively. Students who learn to program the computer can acquire strategic planning and debugging skills, however, programming courses do not necessarily provide direct instruction on these skills.

Programming instruction which focuses only on acquisition of language features and not on how to combine the language features, may fail to foster strategic planning knowledge in students. We can encourage cognitive outcomes from computer learning environments by assigning problems which demand the skills desired. On the other hand, problems which focus primarily on input and output such as reading in "name", printing out "name", and so on are unlikely to require strategic planning. In contrast, problems requiring students to write guessing game programs do require strategic planning. To write a program which gives a player seven chances to guess a number between one and one hundred and gives feedback about whether the number guessed is higher or lower than the number the computer has in mind, requires a series of decisions. The programmer must decide when to generate the computer number, how to keep track of the number of guesses, when to give feedback to the player, etc. Problems which demand strategic planning and which are complex enough that they require students to do debugging increase the likelihood that students will acquire higher cognitive skills from computer learning environments. Activities which lead up to problems which demand strategic planning form the chain of cognitive consequences of computer learning environments.

Instruction can make explicit the skills required for strategic planning and debugging, thus fostering movement along the chain of cognitive consequences. The ACCCEL project has devised a series of techniques for explicitly encouraging planning and debugging. For example, instruction which requires students to make an action diagram of their program prior to writing code encourages students to think about planning. In addition, instruction which requires that students describe at least two hypotheses to explain a bug in their program, before trying to fix it, encourages students to develop debugging skill. In contrast, educational programs which allow students to compose their programs at the keyboard may result in students writing what has been called "spaghetti code", that is, code which is totally unplanned. Educational programs which pay no attention to debugging may encourage haphazard substitution of new code rather than analysis of what went wrong. In the section on teacher professional development, we discuss in greater detail the advantages and disadvantages of explicit instruction for development of higher cognitive skills such as strategic planning.

Pre-college education courses can foster cognitive outcomes by providing an introduction to the information which students might use in computer-related careers and computer-based problem solving in college courses. The problem-solving skills of strategic planning are likely to figure strongly in this instruction. In addition, expert computer problem solvers have developed a repertoire of effective templates for solving problems in their subject matter area. For example, they develop "sorting" templates. Introductory computer courses can and should provide an introduction to these templates. As students gain a repertoire of templates, such courses can also help students develop criteria for selecting an appropriate template for a problem. If the ability to select an appropriate template for a problem is emphasized, it is more likely that each additional computer learning course will allow students to advance along the chain of cognitive consequences.

The cognitive consequences of programming might include generalization to non-computer environments. It has often been pointed out that transfer from one learning environment to another is infrequent and, at best, difficult to accomplish. Recently, researchers have identified meta-reasoning skills as possible subject matter and context independent problem-solving skills used by students. Strategic planning can be seen as one of these skills. Strategic planning is often acquired in a specific subject matter area. It is not uncommon for students to have difficulty discerning which information is subject matter-specific and which is subject matter-independent. Only after a long series of problem-solving experiences do students start to recognize which skills are independent of subject matter and which are tied to subject matter. Courses which provide many problem-solving opportunities and help students recognize their meta-reasoning skills have the potential to enhance problem-solving skill in new domains.

In summary, optimal cognitive consequences from computer learning environments are frequently prevented because instruction fails to emphasize them. School-based planning can change this situation. Much of computer instruction involves drill and practice, which rarely emphasizes higher order problem-solving skills. Even instruction in programming, which has the potential for emphasizing higher order skills, frequently fails to do so. Much of computer programming instruction, as it is currently practiced, focuses on language features rather than on how to combine those features to solve problems. Thus, to enhance the cognitive consequences from computer environments for learning, computers must be used for solving complex problems and instruction must be geared toward planning and debugging rather than toward acquisition of language features.

Implementing a plan. Planning for effective computer-based instructional programs in schools involves specification of desired outcomes in a chain of cognitive consequences including the impact of computer-based programs on other portions of the school curriculum. It is useful to consider, for example, how courses such as typing and mathematics might be affected by an increase in the number of pupils who have computer expertise. Although such forecasting is admittedly

difficult, it is required in order for schools to have effective responses to this important educational innovation.

Effective planning for cognitive outcomes from computer learning also requires coping with the impediments discussed above. Decisions about hardware selection, software selection, and curriculum design often seem fraught with difficulty. Hardware selection depends on the products on the market. The available hardware in the area of computer education is rapidly changing. If one waits for the price to come down or for the best product to be developed, one will wait forever. Decisions have to be made based on what is currently available. Sometimes it is frustrating to make these decisions, however, they cannot be avoided. Computer learning can occur on a variety of available microcomputers as well as larger computers. It is desirable to select hardware which runs the software that you most want to use in your program and not to assume that "promised" software will necessarily be available later on. Decisions have to be made using the hardware and software currently available and assuming that although the situation will change it is not possible to anticipate the direction of that change very effectively.

The selection of software for computer-based environments should reflect the cognitive consequences which are desired. Currently, only a small portion of the available curriculum materials emphasizes strategic planning and debugging, which seem to be among the most unique and valuable cognitive consequences from computer learning environments.

Planning for Affective Consequences

Computer learning environments have the potential for influencing affective outcomes as well as cognitive outcomes. A diverse group of affective consequences of the computer learning environment must be considered in school-based planning.

First, computer education programs can help overcome apprehension sometimes associated with computers. Some students express apprehensions about breaking the computer or concern that the computer might gain control of them in some sense. For almost all learners, this apprehension is overcome quickly because it stems from lack of complete knowledge of the environment. For a few learners, more comprehensive programs such as those developed by the EQUALS program at the Lawrence Hall of Science are needed.

Second, computer education programs can build self-esteem and confidence in one's ability to use computers. Programs which provide the opportunity to use higher cognitive skills have a greater potential for fostering such affective consequences. Recently, we observed a 12-year-old girl solving a complex graphics problem while muttering to herself, "I'm so smart, I can do this." The interactive nature of the environment allowed her to see and correct her mistakes and to recognize that she was making progress on a non-trivial problem.

Third, students can gain confidence and skill in learning autonomously from the computer environment. Much of student learning occurs autonomously, but often it is inefficient and ineffective. The opportunity for interaction and precise feedback when using the computer allows learners to engage in productive autonomous learning, to succeed on difficult problems, and ultimately to gain confidence that they can solve complex problems on their own.

Evidence for autonomous learning comes from the many junior high and high school students who become "expert" programmers, even though courses available to these students typically provide only an introduction to the subject. In spite of this limited exposure, a reasonable number of students learn to program complicated problem solutions. Many of these students have computers at home, others have access to computers during afterschool hours at libraries, museums or computer stores. Computers provide a vehicle for students who desire to learn autonomously to achieve a satisfying outcome.

Fourth, well-planned computer learning experiences have the potential of ensuring that computers will be viewed and used as tools to foster humanitarian goals rather than as means to the mechanization of society. Students who collaborate with their peers to solve problems using the computer are likely to see how this tool can help society. Currently, computer classrooms are among the most interactive. When students encounter bugs in their programs, they generally seek help from peers. Often, students work in teams to design and implement problem solutions. If the problems are complex and if the instructor rewards group work, students often learn to value the contributions of their peers. For example, a group of students is far more likely to generate several possible hypotheses for a program bug than is an individual. Although it is difficult to organize a classroom so collaboration is rewarded, the payoff both in problem solving and in "humanizing" the role of the computer makes such action worthwhile.

Implementing a plan. Planning for effective affective outcomes involves considering both the cognitive demands of the learning environment and the desired affective consequences. Apprehension can be overcome most quickly if computers are fun to use. Self-esteem increases when students solve complex problems. Students are likely to learn autonomously only if they are given problems to solve on their own and if the feedback available provides encouragement. Finally, learners are likely to see the value of computers to humanity when they use it as a tool to foster communication between themselves and their peers.

Planning for Equitable Outcomes

A concern in fostering student outcomes from computer learning environments is the equitable distribution of those outcomes. Lack of access to computer instruction is the major factor which prohibits equitable outcomes from computer learning environments. Females and students in low socioeconomic groups are considerably less likely

to gain access to computers than males and those of high socio-economic status (see Tables 2 and 3). Unless this trend is reduced or eliminated, we can anticipate a strong effect on society. Widening the gap between those who have technological competence and those who do not, further limits the learning potential of those who fail to gain technological expertise. The inequitable distribution of computer learning experiences is already a serious problem; steps must be taken to alleviate it.

Data on the participation of males and females in computer learning environments are reminiscent of the situation in advanced mathematics courses which was publicized in the late 1960's. In California, of the 27,378 students enrolled in computer programming classes in high school, only 37 percent of those students are female. Researchers report that less than 30 percent of the students at computer camps are female. The Lawrence Hall of Science reports that approximately 27 percent of students enrolling in computer courses are female. Female participation in computer learning experiences is much lower than that of males. By requiring courses in computer learning, schools can help to alleviate the situation. By encouraging females to participate in computer learning experiences, families can address this important concern.

The advantages of gaining technological expertise have been well documented both in the area of potential income and in the area of access to further educational experience. Before computer learning experiences become a barrier for future participation in mathematics and science courses and in certain high paying careers, this trend of unequal participation of males and females in computer courses must be reversed.

Implementing a plan. Planning for computer learning environments must include an important emphasis on equitable student outcomes, including increased opportunities for those who failed to get them in early schooling. As the statistics described above point out, the inequality of access to computer learning is already a major problem. Planning efforts must include procedures which insure equitable participation of males and females and of those from high and low socioeconomic groups in computer education programs. Some of this planning can occur at the school district level, other important planning must occur at the state and federal level in order to insure that both districts with high socioeconomic status as well as those with low socioeconomic status provide equitable computer learning opportunities.

Planning Effective Teacher Professional Development

A major problem in the area of computer environments for learning is the lack of training available to teachers in such environments. Even when such training is available, very frequently school districts lack funds for it. Teachers are frequently asked to teach computer-based courses with little or no lead time for planning, minimal training and minimal resources.

Table 2

Participation of Males and Females in

Computer-Related Activities

Enrollment in High School Computer Classes in California

Class	Number Enrolled	Percent Female
Word Processing	1,024	87
Data Processing	800	56
Computer Literacy	9,044	43
Computer Programming	27,278	37
Other Courses Combined	13,335	47
Total	51,481	42

Enrollment in Lawrence Hall of Science Computer Classes, 1981-1983

Class	Number Enrolled	Percent Female
Micros for Micros (Games)	171	37
Creative Play (Games)	600	30
Beginning BASIC	998	26
Pilot	284	33
Intermediate BASIC	378	19
Other Courses Combined	262	23
Total	2,693	27

Majors in Computer Science at U.C. Berkeley 23% female.

Table 3

Availability of Microcomputers*

by Socio-Economic Status (SES)

	Predominantly White			Predominantly	All
	High SES	Middle SES	Low SES	Minority Elementary	
Median % of students who use microcomputers in a week	24	22	12	13	16
Median minutes of use per week of student user	24	22	35	18	23
Percent reporting "intensive" use of microcomputers					
for drill and practice	13	18	9	33	18
for programming	21	17	49	10	23
with "above average" students	24	30	51	26	31
with "average" students	14	9	22	12	12
with "below average" students	16	12	10	32	14

* Taken from School uses of microcomputers, Issue 3, October, 1983, Center for the Social Organization of Schools, The Johns Hopkins University, Baltimore, Md.

Although the majority of teachers providing computer learning environments lack high levels of expertise in computer education, there is a core of expert teachers and their skills must be capitalized upon. Rather than having one teacher after another reinvent excellent procedures for teaching computer programming and problem solving, it is essential that expertise that has been developed be shared. Many successful professional development programs for teachers capitalize on helping expert teachers make their expertise explicit and share it with other teachers. For example, the Bay Area Writing Project and the Bay Area Mathematics Project bring teachers together, have them share their ideas, and start to make their expertise explicit.

Recent advantages in psychological and educational research have, in many cases, been consistent with the ideas of thoughtful and expert teachers. When teachers are asked to make their knowledge explicit, the relationships between their knowledge and the findings of recent research can be articulated. Research findings combined with the knowledge of expert teachers can be used to develop effective professional development programs.

An example of the interface between teacher expertise and recent research occurs in the area of explicit procedures for problem solving. Teachers frequently employ discovery learning procedures rather than explicit instruction in problem solving, although critics sometimes claim that discovery learning is too time consuming and that explicit instruction for solving the problems would be more effective. The preference for discovery learning frequently voiced by expert teachers is reflected in recent research. This research reveals that explicit or direct instruction tends to have value for some learners. Explicit instruction apparently helps learners who have no clues as to how to proceed in a given problem-solving situation and who require considerable instructional support in order to solve problems. These learners are usually of lower general ability than learners who respond well to discovery learning. In addition, they frequently have less experience with the subject matter domain than those who perform well in the discovery learning mode.

On the other hand, students who profit from discovery learning are generally high ability students who have some knowledge of the subject matter area. These students profit more from discovery learning than from explicit instruction because they are able to devise their own algorithms and heuristics for solving problems. If algorithms or heuristics for problem solving are provided for them, then these instructions may duplicate their own problem-solving process and may dissuade students from developing an "understanding" of how to solve the problems.

Thus, an aptitude treatment interaction where students with high ability are encouraged to learn in a discovery mode while those with low ability are provided with direct instruction is suggested. Results from one of the ACCCEL research studies which investigated student performance on WUMPUS tends to support this point of view. In that study, explicit instruction was helpful for low-ability

learners while some high-ability learners profited from a more discovery-oriented form of instruction.

Thus, teacher professional development can be fostered by capitalizing on the knowledge of expert teachers and by relating that knowledge to recent research advances. Together, these two forms of information can be combined to create better instructional programs than either form would provide on its own.

Teacher professional development can also be fostered by the networking of teachers to share especially effective practices. Networking of teachers has the advantage of encouraging the dissemination of information developed at local sites and freeing teachers from the necessity of reinventing procedures.

Implementing a plan. Teacher professional development must be an important component of any plan for computer learning. It is unreasonable and ineffective to assume that large numbers of teachers will train themselves to use such a complicated tool. When businesses adopt word processing or economic spread sheet software, they provide extensive training for the professionals who will be using those tools. Similarly, teachers deserve extensive training in use of new professional tools.

There is a growing understanding of the process of teacher professional development. Knowledge is emerging at both research universities and local schools. It is necessary to build new models for linkages between schools, universities and research agencies which foster effective school practice. For example, the relationship between teachers' perceptions of discovery learning and recent research on discovery learning can be investigated by teams representing local school practitioners and researchers.

A Model of Computer-Based Instruction

Both the ideas of expert teachers and the results of research on computer learning environments are beginning to suggest a model of computer-based instruction which could have far-reaching consequences. Such a model is needed to clarify how instruction can capitalize on the unique features of computers. For example, the emphasis on explicit instruction in conjunction with discovery learning becomes far more central in educational decision making using computers than it does in other aspects of the curriculum. The desired effect from explicit instruction is frequently not accomplished in non-computer-based learning environments simply because of the difficulty involved in implementing it. Computers provide the opportunity for explicit instruction and also provide the necessity of making decisions about whether explicit instruction is appropriate for all learners.

Computer learning environments provide the opportunity for emphasizing aspects of the curriculum that might be difficult or impossible to emphasize without computers. For example, strategic

planning knowledge is rarely taught in regular classrooms, simply because it requires a great deal of student and teacher interaction. Computer environments provide numerous subject matter areas where strategic planning knowledge might be encouraged and where students can gain explicit and direct feedback on the effectiveness of their strategic planning. Even when strategic planning is required in non-computer learning environments (such as in the construction of geometry proofs), feedback may be lacking in explicitness or may occur a long time after the problem has been attempted. Computer environments provide opportunities for more explicit and more immediate feedback.

Important contributions from both research universities and local schools are necessary for the development of a model of computer-based instruction. The input of expert teachers, as well as the results of recent research such as that conducted by the ACCCEL project, need to be combined in order to form an effective model. Computer-based instruction provides an especially exciting opportunity for this interaction.

Conclusions

The potential of computers in education provides a rare opportunity for educators. The computer can greatly increase the likelihood that the educational reforms called for by recent national commissions will be achieved. To accomplish this, we must narrow the gap between the promise of computers in education and the reality of their use in schools.

School-based planning can increase the effectiveness of computer learning environments. Unless schools respond to the gap between promise and reality in computer education, the gap will widen. The impediments to effective computer education will increase unless schools demonstrate that they want to use computers to their full potential. If schools demonstrate that they want cognitively demanding software and integrated curricula, if schools develop the ability to select effective software, and if schools seem to be making informed rather than haphazard decisions about hardware and software, then developers will be more eager to develop products for the educational market.

Schools and researchers can work together to gain more understanding of the potential of the computer learning environment, to clarify desired outcomes for students, and to establish the chain of cognitive consequences from computer learning environments. Expert teachers are designing programs to capitalize on computers. Research programs such as ACCCEL are conducting investigations of the chain of cognitive consequences, examining the responses of males and females to computer learning environments, and gathering other information useful to educators. If these groups work together, planning can be simplified.

To capitalize on the potential of computers in education, resources must be allocated effectively. The temptation to purchase hardware rather than investing resources in planning and in teacher professional development may impede optimum program development. The issues surrounding implementation of computer education programs are complex; they deserve serious and thoughtful consideration. The costs of planning and teacher professional development are small relative to their potential benefits.

Ultimately, effective use of computer learning environments requires the design of hardware, software, and curriculum materials focused exclusively and specifically on the problems of schools. The development of such a response requires the cooperation of schools, state governments, federal governments, and research agencies. Without the effective integration of all of these agencies, we can continue to expect to see sporadic and incomplete efforts in computer learning and we can continue to expect to hear about the gap between the promise of computer learning environments and the reality of instructional use of computers in schools. The macro expectations for computers in education will be met with micro outcomes unless thoughtful individuals and substantial resources are devoted to this problem.

The potential for the use of computers in the instructional environment is phenomenal. It would be irresponsible to ignore this potential. Substantial, serious, and concerted effort is needed in order to incorporate this new and effective educational tool into creative, innovative, and extensive educational programs.